SeerNet@INLI-FIRE-2017: Hierarchical Ensemble for Indian Native Language Identification

Royal Jain
Venkatesh Duppada
Sushant Hiray
royal.jain@seernet.io
venkatesh.duppada@seernet.io
sushant.hiray@seernet.io
Seernet Technologies, LLC
Milpitas, CA, USA

ABSTRACT

Native Language Identification has played an important role in forensics primarily for author profiling and identification. In this work, we discuss our approach to the shared task of Indian Language Identification. The task is primarily to identify the native language of the writer from the given XML file which contains a set of Facebook comments in the English language. We propose a hierarchical ensemble approach which combines various machine learning techniques along with language agnostic feature extraction to perform the final classification. Our hierarchical ensemble improves the TF-IDF based baseline accuracy by 3.9%. The proposed system stood 3rd across unique team submissions..

CCS CONCEPTS

• **Computing methodologies** → **Classification and regression trees**; **Support vector machines**; **Neural networks**; **Bagging**; **Feature selection**;

KEYWORDS

Native Language Identification, Text Classification, Ensemble

1 INTRODUCTION

Native Language Identification (NLI) is primarily the task of automatically identifying the native language of an individual based on their writing or speech in another language. The underlying assumption here is that an author's native language (mother tongue) will often have an influence on the way they express themselves in another language. Identifying such common patterns across a group of people can be used to determine their native language.

Identifying the native language of an author has various applications, primarily in forensics. In forensics, author profiling and identification using their native language is an important feature [1]. Identifying the native language can also be used to provide personalised training for learning new languages [6]. Recent work by [3] focuses on using this in tracing linguistic influences in multi-author texts.

Researchers have experimented with a range of machine learning algorithms, with Support Vector Machines having found the most success. However, some of the most successful approaches have made use of classifier ensemble methods to further improve performance on this task.

In this shared task [2] we focus on identifying the native language for users from their comments on various Facebook news posts. From Natural Language Processing (NLP) perspective, NLI is framed as a multiclass supervised classification task. The shared task at hand is specific to identifying six Indian native languages: Tamil, Hindi, Kannada, Malayalam, Bengali and Telugu.

As we explore in the next section, prior work has primarily dealt with statistical machine learning algorithms including SVMs and representation methods such as tf-idf. Our approach combines these various state of the art algorithms using a hierarchical ensemble. We've also experimented with two different types of feature extraction strategies. They are explored further in Section 3.1

2 RELATED WORK

Most of the related NLI work can be categorized into 2 domains: text based and speech based.

2.1 Text NLI

The 2013 Native Language Identification Shared Task [8] created an increased interest in the problem by providing a large labelled dataset. [9] exploited difference in parse structure in texts of different native language speakers for reducing classification error. Very recently, the 2017 shared task on Native Language Identification [4] provided additional contributions to the field.

2.2 Speech NLI

[10] demonstrates that the acoustic features along with various features computed on the transcripts can provide increased accuracy in dialect identification. [7] achieved good results with i-vector and glove vector features with a GRU deep learning model.

Starting from the shared task in 2013, quite a few approaches used ensembling techniques to combine multiple base classifiers to improve the performance.

3 SYSTEM DESCRIPTION

3.1 Feature Extraction

We observe from the dataset that people often use words and phrases which belong to their native language transliterated into English. Some common examples are "Jai ho", "vadi koduthu" etc. We also expect that people who have the same native language would have some topics/concerns which would not be shared by

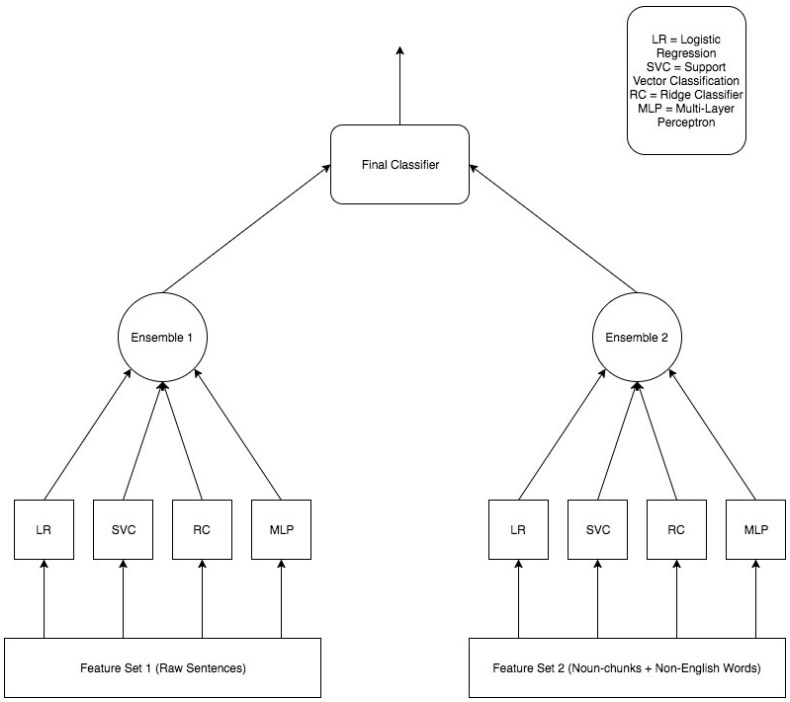

Figure 1: System Design

people who have a different native language. For example an issue which revolves around Tamil Nadu would resonate more with Tamil speaking people as compared to others.

For our classification system we created two different feature sets from our data. In the first feature set we take raw sentences as inputs. The sentences are tokenized to create vocabulary of tokens. This vocabulary is then used to create term frequency-inverse document frequency features for each sample point which are then used as feature input in the classification step. One benefit of term frequency inverse document frequency over simple bag of words approach is it mitigates the effect of common words and thus making inputs easier to discriminate. We refrained from using higher n-grams features due to limited amount of data.

In the second feature set we leverage our observations stated above to filter the relevant information. First, for each sentence we collect words which do not belong to the English vocabulary. The sentences were tokenized using tweetokenize package[1] and we check whether the word belongs to English vocabulary by using the English dictionary provided in enchant [2]. These words are extracted for capturing usage of native language in inputs. We then tested our hypothesis that speakers of common native language would have topics/concerns which are not shared as strongly by others. To this end we collected all the documents of native speakers of each language and extracted topics from it using Latent Dirichlet Allocation. We observed a good deal of topics which were specific to speakers of common native language. We think this is a result of regional and cultural proximity between speakers of common native language. Most of these topics were expressed in the noun forms and hence to extract this information we collected noun chunks which are present in the sentences. Noun chunks are extracted using spacy[3]. Now we follow procedure similar to first feature set. We collect these two features for each sentence and then create a vocabulary for it. This vocabulary is then used to create term frequency inverse document frequency features which are then used as inputs for classification.

3.2 Classification

We perform the classification separately for both feature sets described above. The training data set in the competition was small hence, instead of creating separate train and development set, we performed 10-fold cross validation. On each fold, a model was trained and the predictions were collected on the remaining dataset. We calculated mean of accuracy over 10 fold for each type of classifier. We also observed the performance of each classifier on points which were harder to classify i.e those points for which the decisions were incorrect for majority of classifiers. After evaluation selected four classifiers, namely LogisticRegression, MLPClassifier, LinearSVC and RidgeClassifier of sklearn [5], were selected for ensemble creation. These classifiers were chosen based on their performance on the cross-validation and also on the basis of their complimentary performance on hard to predict data points. The performance of these classifiers on cross validation is shown in table 1.

[1]https://www.github.com/jaredks/ tweetokenize
[2]https://pypi.python.org/pypi/pyenchant/
[3]https://spacy.io/

Table 1: 10-fold Cross Validation Mean Accuracy on feature sets

Classifier	Feature Set1	Feature Set2
LogisticRegression	0.887959046018	0.912401033115
LinearSVC	0.894482995578	0.914853592818
RidgeClassifier	0.894476444138	0.913260049508
MLPClassifier	0.878357282545	0.902736002619

Table 2: Accuracy on test data

Class	Submission1	Submission2	Submission3
BE	64.40	64.80	67.10
HI	16.10	14.30	15.70
KA	49.80	46.50	48.10
MA	46.80	50.00	45.40
TA	54.40	52.10	52.20
TE	44.40	43.70	44.90
OverAll	46.60	46.40	46.90

3.3 Ensemble

We created a hierarchical ensemble model for this task, consisting of two layers of ensembles. First layer consists of two ensemble. First one consists of four classifiers selected in the previous section mentioned. These classifiers were trained on feature set 1 (term frequency inverse document frequency features on raw input sentences). Second ensemble also consists of same four classifiers but were trained on feature set 2, which had term frequency inverse document frequency features computed using noun chunk and non English words extracted from each sentence. Each ensemble predicts the output using the majority vote. We limited the decision to majority vote as complex weighted voting would have caused overfitting. Final classification is predicted using a combination of two ensembles described above. If they output same class, we present that class as prediction. If they differ, we calculate the confidence of each ensemble using count of classifiers in the ensemble which support its decision. Fig 1. depicts our system.

4 RESULTS

We can see from table 1. that all four classifiers perform quite well on both the extracted feature sets especially considering the classification problem involves six classes. This suggest that the dataset points are easier to discriminate. We further see that the accuracy increases significantly on feature set 2, suggesting that features such as native language words and regional/local topics are important for identification of native language.

We presented three submissions. Submission 1 is the output of final classifier(see Fig 1). Submission 2 is the output of Ensemble 1, which was trained on raw sentences. Submission 3 was generated using ensemble 2 trained on feature set 2 (non-English phrase and noun chunks). We can see that Submission 3 outperform other two classifiers strengthening our belief on importance of native language phrases and shared topics in identifying native language of speaker.

5 FUTURE WORK AND CONCLUSION

This paper studies couple of approaches for identification of native language. First approach measures the power of tf-idf features for the purpose of classification. Second approach identifies certain features which separate different native language speakers from each other and utilizes those for better ac curacies of overall system. We have seen improvement in accuracy due to identification of discriminating features, however extending this procedure is time consuming and requires language expertise. Recent studies have shown use of deep neural networks can be a possible alternate to creating manually hand-crafted features and can provide better performance.

ACKNOWLEDGMENTS

We would like to thank the organisers of the FIRE-2017 Shared Task on Native language identification, for providing the data, the guidelines and timely support.

REFERENCES

[1] John Gibbons. 2003. *Forensic linguistics: An introduction to language in the justice system.* Wiley-Blackwell.
[2] Anand Kumar M, Barathi Ganesh HB, Shivkaran S, Soman K P, and Paolo Rosso. 2017. Overview of the INLI PAN at FIRE-2017 Track on Indian Native Language Identification. In *Notebook Papers of FIRE 2017.*
[3] Shervin Malmasi and Mark Dras. 2017. Native Language Identification using Stacked Generalization. *arXiv preprint arXiv:1703.06541* (2017).
[4] Shervin Malmasi, Keelan Evanini, Aoife Cahill, Joel Tetreault, Robert Pugh, Christopher Hamill, Diane Napolitano, and Yao Qian. 2017. A Report on the 2017 Native Language Identification Shared Task. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications.* 62–75.
[5] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.
[6] Alla Rozovskaya and Dan Roth. 2011. Algorithm selection and model adaptation for ESL correction tasks. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1.* Association for Computational Linguistics, 924–933.
[7] Ishan Somshekar, Bogac Kerem Goksel, and Huyen Nguyen. [n. d.]. Native Language Identification. ([n. d.]).
[8] Joel R Tetreault, Daniel Blanchard, and Aoife Cahill. 2013. A Report on the First Native Language Identification Shared Task.. In *BEA@ NAACL-HLT.* 48–57.
[9] Sze-Meng Jojo Wong and Mark Dras. 2011. Exploiting Parse Structures for Native Language Identification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP '11).* Association for Computational Linguistics, Stroudsburg, PA, USA, 1600–1610. http://dl.acm.org/citation.cfm?id=2145432.2145603
[10] Marcos Zampieri, Shervin Malmasi, Nikola Ljubešić, Preslav Nakov, Ahmed Ali, Jörg Tiedemann, Yves Scherrer, and Noëmi Aepli. 2017. Findings of the VarDial Evaluation Campaign 2017. (2017).